



## Questão 2

Completo

Vale 1,50 ponto(s).

Considere duas variáveis  $f$  e  $i$  associadas aos registradores  $\$s0$  e  $\$s1$ , respectivamente. Considere ainda um vetor  $A$  cujo endereço base está nos registradores  $\$s6$ . Traduza o trecho de código em linguagem C a seguir para assembly MIPS.

```
for (i = 0; i < 10; i++) {  
    A[i] = f + i;  
}
```

.text:

```
move $s0, $zero
```

```
move $s1, $zero
```

- \$2,10

main:

```
li $v0, 0
```

loop:

```
bgt $t0, $t2, exit
```

add \$t1, \$s0, \$s1

SW \$t1, \$s1 (\$s6)

addi \$s1, \$s1, 1

loop

exit:

Questão 3

Completo

Vale 2,00 ponto(s).

Vimos que as instruções em assembly MIPS são traduzidas para linguagem de máquina para poderem ser executadas pelo hardware. Essa tradução consiste em transformar uma instrução e seus operandos em uma palavra de 32 bits. Há 3 formatos de representação para a tradução de instruções.

1. O formato do **tipo R** é representado pelos seguintes campos:

op	rs	rt	rd	shamt	funct
6 bits	5 bits	5 bits	5 bits	5 bits	6 bits

onde op é o código que identifica a operação, rs e rt são os registradores dos operandos, rd é o registrador do destino, shamt é a quantidade de deslocamento (shift amount) e funct é o código da função (complementa o campo op).

2. O formato do **tipo I** é representado pelos seguintes campos:

op	rs	rt	constante ou endereço
6 bits	5 bits	5 bits	16 bits

3. O formato do **tipo J** é representado pelos seguintes campos:

op	constante ou endereço
6 bits	26 bits

Com base nos formatos, responda às seguintes questões.

- a. Para cada um dos três formatos, dê ao menos um exemplo de instrução que use o respectivo formato na sua representação em linguagem de máquina.
- b. Você está cansado da pouca quantidade de registradores que a arquitetura MIPS possui e resolve reprojeter a arquitetura de forma alterar a quantidade de registradores para 128. Com essa modificação, qual o tamanho, em bits, que uma palavra deve ter para representar uma instrução em cada um dos 3 formatos? Justifique sucintamente.

a.

Tipo R:

add \$s1, \$s2, \$s3

Tipo I:

lw \$t0, 32(\$t1)

Tipo J:

j 1000 #va para 1000

b.

Na arquitetura mips tem registradores (célula de memória: palavras) tem tamanho de 32 bits, logo aumentando os registradores de 32 para 128 as palavras também passam a ser de 128 bits.

Tipo R: 128

op	rs	rt	rd	shamt	funct	op	rs	rt	const ou end
24	20	20	20	20	24	24	20	20	64

Tipo J: 128

op	const ou end
24	104

Tipo I: 128

